

Research Article

Terrain Adaptive Estimation of Instantaneous Centres of Rotation for Tracked Robots

Cuifeng Wang¹, Wenjun Lv², Xiaochuan Li³ and Mingliang Mei¹

¹Department of Mechanical Engineering, Fujian Polytechnic of Information Technology, Fuzhou 350003, China

²Department of Automation, University of Science and Technology of China, Hefei 230027, China

³Faculty of Technology, De Montfort University, Leicester LE1 9BH, UK

Correspondence should be addressed to Cuifeng Wang; cfwang@fjpit.edu.cn and Wenjun Lv; wlv@ustc.edu.cn

Received 25 June 2018; Accepted 15 August 2018; Published 1 November 2018

Academic Editor: Zhile Yang

Copyright © 2018 Cuifeng Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

As a type of skid-steering mobile robot, the tracked robot suffers from inevitable slippage, which results in an imprecise kinematics model and a degradation of performance during navigation. Compared with the traditional robot, the kinematics model is able to reflect the influences of slippage through the introduction of instantaneous centres of rotation (ICRs). However, ICRs cannot be measured directly and are time-varying with terrain variation, and thus, here, we aim to develop an online estimation method to acquire the ICRs of a robot by means of data fusion technologies. First, an innovation-based extended Kalman filter (IEKF) is employed to fuse the readings from two incremental encoders and a GPS-compass integrated sensor, to provide a real-time ICR estimation. Second, a decision tree-based learning system is used to classify the terrains that the robot traverses, according to the vibration signals gathered by an accelerometer. The results of this terrain classification are improved via a Bayesian filter, by utilizing temporal correlation in the terrain time series. Third, the performances of the ICR estimation and terrain classification are mutually promoted. On one hand, terrain variation is detected with the aid of the terrain classification, and therefore, the process noise variance of IEKF can be automatically adjusted. Hence, the results of ICR estimation are smooth if the terrain does not change and converge rapidly upon terrain variation. On the other hand, the sudden changes in innovation are used to adjust the state transition probability during the recursive calculation of the Bayesian filter, thus increasing the accuracy of the terrain classification. A real-world experiment was undertaken on a tracked robot to validate the effectiveness of the proposed method. It is also demonstrated that the terrain adaptive odometry outperforms the traditional approach with the knowledge of ICRs.

1. Introduction

Tracked robots are able to govern their headings by adjusting the relative speed between the left and right driving wheels rather than resorting to independent steering mechanisms. Due to their robustness, the simplicity, and the ability of zero-radius rotation, tracked robots have become the preferred all-terrain robots in agriculture, industry, and military. However, due to the presence of the large contacting patches between the tracks and the ground, unpredictable slip is inevitable, which makes it difficult to build a precise kinematics model [1]. Because most strategies that concern navigation, motion control, obstacle avoidance, and route planning are designed based on the kinematics model, a robot's performance can be significantly improved by the online

identification of terrain-related slip parameters included in the kinematics model [2–8]. Therefore, this paper concentrates on designing a data fusion approach to acquire slip parameters in real time.

Due to fact that the instantaneous centres of rotation (ICRs) of the chassis and tracks are almost constant when the robot is moving on the same terrain, the ICR kinematics model (as illustrated in Figure 1) was proposed by introducing ICRs into the traditional kinematics model of a difference-driven mobile robot [9]. The ICR locations are able to reflect a robot's lateral and longitude slippage, so they can be considered as a kind of slip parameters, and therefore, we are in the position to develop an ICR estimation method. In [9], two different methods were presented for the offline estimation of ICRs: the stationary response simulation of the dynamics

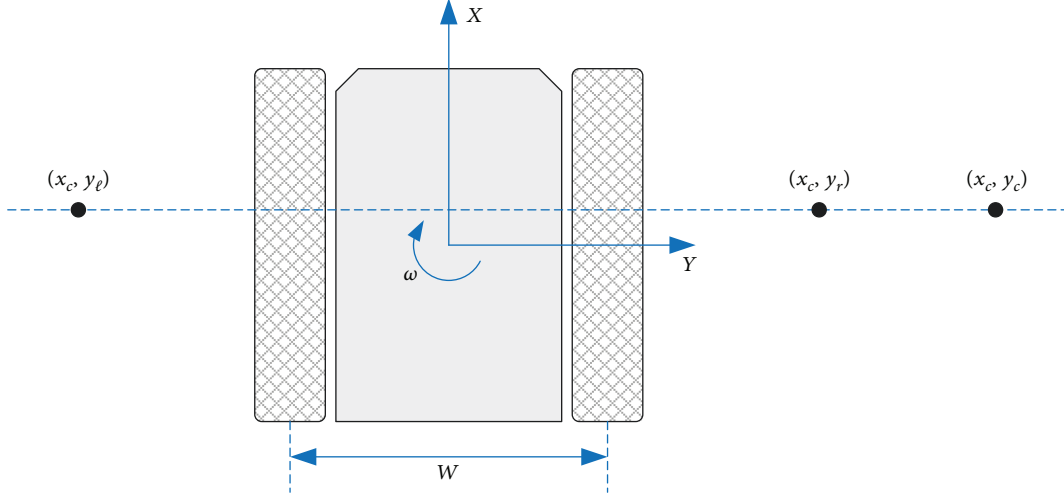


FIGURE 1: Schematic of the ICR locations of a tracked robot.

model within the whole range of velocity, and the genetic algorithm based model parameters extracted from the gathered readings of sensors. In [10, 11], an empirical model combining ICRs and a robot's kinematics state (i.e., the forward speed and radius of the path curvature) was established experimentally, which has been proven to improve the performance of dead reckoning significantly. Similar work in [12, 13] has been described. The primary issue of these off-line methods is that the robot-terrain mapping database should be established in advance, so that the ICR estimation cannot proceed when the robot enters the terrains which are not included in this database. Therefore, the current methods are converted to assist in the development of on-line ICR estimation methods.

Because the states of an ICR kinematics model are composed of the robot's pose and ICRs, we can estimate the ICRs by means of pose measurements, which are implemented by fusing two incremental encoders and a GPS-compass integrated sensor. In [14], an extended Kalman filter (EKF) was employed to realize the ICR estimation, and its effectiveness was verified by a number of experiments in which the robot traverses terrains in different motion patterns. With a more precise kinematics model, the robot's trajectory can be accurately calculated by incorporating the readings from on-board sensors into the model [15]. However, due to the uncertainties in the ICR kinematics model and the linearization-induced errors produced in EKF, the EKF-based ICR estimation method may lose accuracy or even become unreliable. The sudden terrain change and variation kinematics states may result in the degradation of system performance as well. In [16], a strong tracking filter (STF) and standard Kalman filter were applied to estimate the robot's kinematics states and ICRs, respectively. By introducing suboptimal fading scaling factors, the filtering gain can be adjusted to guarantee the orthogonality of the innovation series. Hence, this method has a better state tracking ability and higher robustness against model uncertainties and state jump. The work was also accomplished by using a sliding mode observer and error-tolerant switched robust filter [17, 18].

To the best of our knowledge, the aforementioned research represents state-of-the-art technologies concerning

ICR estimation. The existing studies focus on the fusion approach, such as the Kalman filter and its extensions, which commonly suffer from the following two issues. The process noise covariance of ICRs should be set as a relatively large constant to enable the rapid convergence of ICR estimation, but the ICR estimation results will oscillate significantly around the true values. On the contrary, to guarantee smoothness, the process noise covariance should be set to a relatively small constant at the cost of slow convergence. Hence, the setup of the process noise covariance faces the inevitable trade-off between the convergence rate and the smoothness of the ICR estimation. Second, due to the model uncertainties rendered by linearization, parameter variation, and the measurement variance change, the filter-computed filtering gain may not maintain the optimality, so an adaptation mechanism should be introduced to automatically adjust the filtering gain.

In this paper, we propose a solution to tackle these issues by using the terrain-adaptive innovation-based extended Kalman filter (TA-IEKF). First, an innovation-based Kalman filter (IKF) is employed to fuse the readings from two incremental encoders and a GPS-compass integrated sensor, to provide a real-time ICR estimation. Second, a decision tree-based learning system is utilized to classify the terrain over which the robot is traversing, according to vibration signals gathered by an accelerometer. The results of the terrain classification are improved via a Bayesian filter, by utilizing temporal correlation in the terrain time series. Third, the performances of the ICR estimation and terrain classification are mutually promoted. On one hand, any terrain variation is detected with the aid of terrain classification, and therefore, the process noise variance of the IEKF can be automatically adjusted. Hence, the results of the ICR estimation are smooth if the terrain does not change and converge rapidly upon terrain variation. On the other hand, the sudden changes in innovation are used to adjust the state transition probability during the recursive calculation of the Bayesian filter, thus increasing the accuracy of the terrain classification.

The remainder of the paper is organized as follows. Section 2 presents the ICR kinematics model. Section 3

describes the details of the proposed terrain adaptive innovation-based extended Kalman filter. Section 4 presents a real-world experiment and an analysis of its results. A conclusion is provided in Section 5.

2. ICR Kinematics Model

The section presents the ICR kinematics model for a tracked robot. As shown in Figure 1, the body coordinate system (BCS) is assumed to have its origin on the geometric centre of robot body. The X -axis and Y -axis are aligned with the longitudinal forward direction and lateral rightward direction, respectively. The ICR location of the robot body on the Y -axis, y_c , is

$$y_c = \frac{v_x}{\omega}, \quad (1)$$

where v_x denotes the robot velocity along the X -axis and ω denotes the turning rate. The ICR locations of the left and right tracks on the Y -axis, y_ℓ and y_r , are

$$y_\ell = \frac{v_x - v_\ell^x}{\omega}, \quad (2a)$$

$$y_r = \frac{v_x - v_r^x}{\omega}, \quad (2b)$$

where v_ℓ^x and v_r^x denote the left- and right-track velocities along the X -axis, respectively. The ICR location of the robot chassis on the X -axis, x_c , is

$$x_c = -\frac{v_y}{\omega}, \quad (3)$$

where v_y denotes the robot velocity along the Y -axis. The ICR locations of the left and right tracks on the X -axis, x_ℓ and x_r , are

$$\begin{aligned} x_\ell &= -\frac{v_y - v_\ell^y}{\omega}, \\ x_r &= -\frac{v_y - v_r^y}{\omega}, \end{aligned} \quad (4)$$

where v_ℓ^y and v_r^y denote the left- and right-track velocities along the Y -axis, respectively.

It is assumed that the rotating tracks do not produce the track motion along the Y -axis, so $v_\ell^y = v_r^y = 0$, and therefore, we have

$$x_c = x_\ell = x_r = -\frac{v_y}{\omega}. \quad (5)$$

Hence, in the ICR locations of the chassis, the left and right tracks are on the same line parallel to the Y -axis.

As seen in (1), when the robot moves in a straight line, i.e., $\omega = 0$, y_c ranges within $[-\infty, +\infty]$. However, the numerators and denominators of (2a), (2b), and (3) are infinitesimals of the same order, which result in bounded values of y_ℓ , y_r , and x_c [9].

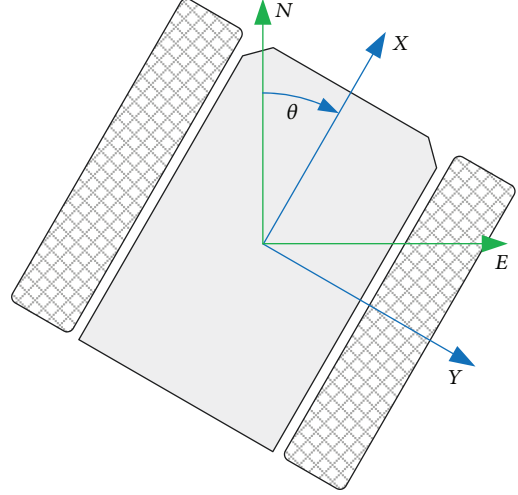


FIGURE 2: Coordinate systems on a tracked robot. The blue axes represent the body coordinate system, and the green ones represent the world coordinate system.

Solving the velocities with respect to BCS, i.e., v_x , v_y , and ω , from (2a), (2b), and (3), we have

$$\begin{aligned} v_x &= \frac{v_r y_\ell - v_\ell y_r}{y_\ell - y_r}, \\ v_y &= \frac{(v_\ell - v_r) x_c}{y_\ell - y_r}, \\ \omega &= -\frac{v_\ell - v_r}{y_\ell - y_r}, \end{aligned} \quad (6)$$

the denominators of which are nonzero values. Here, since $v_\ell^y = v_r^y = 0$, the superscripts of v_ℓ^x and v_r^x are discarded, i.e., $v_\ell = v_\ell^x$ and $v_r = v_r^x$. The ICRs lie on or outside the tracks' central lines when slippage does or does not occur [10]. Hence, $y_\ell - y_r \geq -W$ where W denotes the robot width.

Now, we are in the position to transform the kinestates with respect to BCS to those with respect to the world coordinate system (WCS). As illustrated in Figure 2, the E -axis and N -axis of WCS are eastward and northward, respectively. To establish the ICR kinematics model, the following assumptions are made:

- (i) All tracks are in contact with the ground, so the robot does not separate from the ground
- (ii) The moving plane is planar, so the height, pitch, and roll are ignored
- (iii) All the driving wheels are the same in size, and they do not suffer from deformation

The kinestates with respect to BCS and WCS can be related by

$$\begin{aligned} \dot{e} &= v_x \sin \theta + v_y \cos \theta, \\ \dot{n} &= v_x \cos \theta - v_y \sin \theta, \\ \dot{\theta} &= \omega, \end{aligned} \quad (7)$$

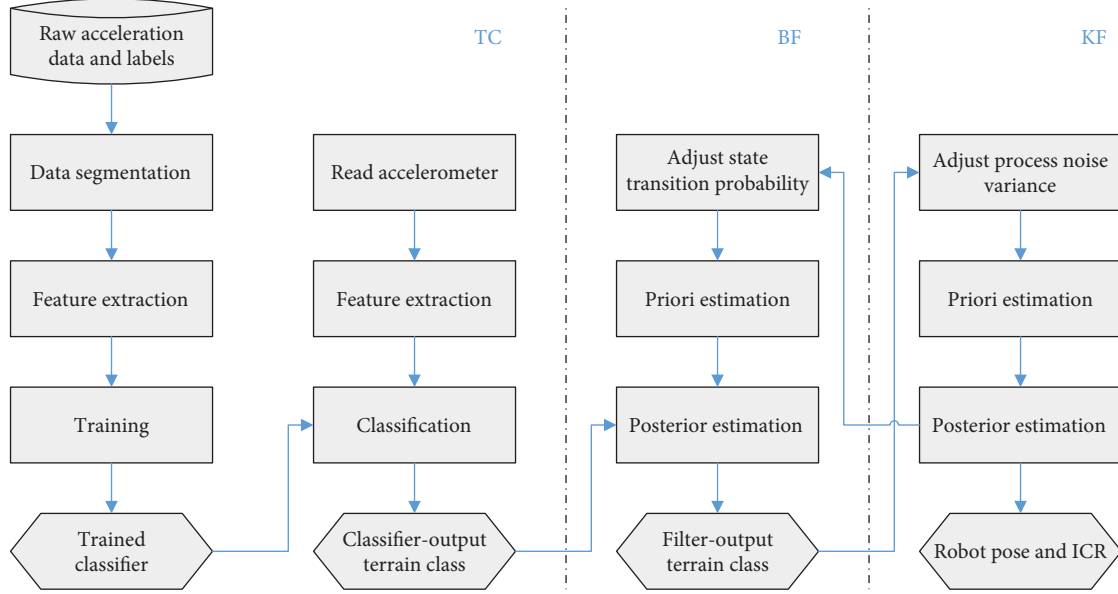


FIGURE 3: Schematic of the proposed terrain adaptive innovation-based extended Kalman filter. TC stands for terrain classification, BF stands for Bayesian filter, and KF stands for Kalman filter.

where e and n denote the eastward and northward location coordinate with respect to WCS and θ denotes the heading angle which is defined by the angle from the N -axis and the X -axis.

3. Terrain Adaptive IKF

The schematic of the proposed ICR estimation based on TA-IEKF is illustrated in Figure 3. For the ICR estimation, we assign a relatively small value to the process noise variance of ICR, thus to guarantee the estimation smoothness. At the moment that the terrain changes, a relatively large value is assigned to the process noise variance of the ICR, thus accelerating the estimation of the convergence. After obtaining several sampling points, the previously assigned value is reassigned to the process noise variance of ICR. Therefore, the smoothness and rapid convergence of the ICR estimation can both be guaranteed.

In order to identify the event at which the terrain changes, we resort to vibration-based terrain classification. First, the vibration signals for different terrains are gathered by using an accelerometer. After the data segmentation, feature extraction, and model training, the classifier is obtained, and therefore, the terrain on which the robot is moving can be recognized in real time. Among various classification methods, the efficiency is the primary factor to be considered, because a tracked robot prefers low energy consumption and low computational complexity. Second, the lost accuracy rendered by the employment of simple features and a weak classifier is compensated by using a Bayesian filter, based on the temporal correlation in the terrain time series. To guarantee a higher filtering accuracy, the diagonal elements of the state transition probability matrix should be set at a value approximating

to 1. However, massive errors occur at the moment that the terrain changes. Fortunately, a sudden increase in innovation is read as a sign indicating terrain variation, and consequently, the state transition probability matrix can be adjusted to reduce the error rate. It is noted that a sudden change in innovation cannot be used to activate the adjustment in the process noise variation of ICR, because it is a rough indicator of the terrain variation. A wrong indicator does not give rise to a large error in terrain classification, but a significant degradation in ICR estimation.

$$\begin{bmatrix} e_{t+1} \\ n_{t+1} \\ \theta_{t+1} \\ y_{\ell,t+1} \\ y_{r,t+1} \\ x_{c,t+1} \end{bmatrix} = \begin{bmatrix} e_t + T \left(\frac{v_{r,t} y_{\ell,t} - v_{\ell,t} y_{r,t}}{y_{\ell,t} - y_{r,t}} \sin \theta_t + \frac{x_{c,t} (v_{\ell,t} - v_{r,t})}{y_{\ell,t} - y_{r,t}} \cos \theta_t \right) \\ n_t + T \left(\frac{v_{r,t} y_{\ell,t} - v_{\ell,t} y_{r,t}}{y_{\ell,t} - y_{r,t}} \cos \theta_t - \frac{x_{c,t} (v_{\ell,t} - v_{r,t})}{y_{\ell,t} - y_{r,t}} \sin \theta_t \right) \\ \theta_t + T \left(\frac{v_{r,t} - v_{\ell,t}}{y_{\ell,t} - y_{r,t}} \right) \\ y_{\ell,t} \\ y_{r,t} \\ x_{c,t} \end{bmatrix} + T \cdot \begin{bmatrix} w_{e,t} \\ w_{n,t} \\ w_{\theta,t} \\ w_{\ell,t} \\ w_{r,t} \\ w_{c,t} \end{bmatrix}, \quad (8)$$

$$F_{\mathcal{K},t} = \begin{bmatrix} 1 & 0 & T \left(\frac{v_{r,t} \hat{y}_{\ell,t} - v_{\ell,t} \hat{y}_{r,t}}{\hat{y}_{\ell,t} - \hat{y}_{r,t}} \cos \hat{\theta}_t + \frac{\hat{x}_{c,t} (v_{r,t} - v_{\ell,t})}{\hat{y}_{\ell,t} - \hat{y}_{r,t}} \sin \hat{\theta}_t \right) \\ 0 & 1 & T \left(\frac{v_{\ell,t} \hat{y}_{r,t} - v_{r,t} \hat{y}_{\ell,t}}{\hat{y}_{\ell,t} - \hat{y}_{r,t}} \sin \hat{\theta}_t + \frac{\hat{x}_{c,t} (v_{r,t} - v_{\ell,t})}{\hat{y}_{\ell,t} - \hat{y}_{r,t}} \cos \hat{\theta}_t \right) \\ 0 & 0 & 1 \end{bmatrix}, \quad (9a)$$

$$F_{\mathcal{S},t} = T(v_{\ell,t} - v_{r,t}) \begin{bmatrix} \frac{\hat{y}_{r,t} \sin \hat{\theta}_t}{(\hat{y}_{\ell,t} - \hat{y}_{r,t})^2} - \frac{\hat{x}_{c,t} \cos \hat{\theta}_t}{(\hat{y}_{\ell,t} - \hat{y}_{r,t})^2} & \frac{-\hat{y}_{\ell,t} \sin \hat{\theta}_t}{(\hat{y}_{\ell,t} - \hat{y}_{r,t})^2} + \frac{\hat{x}_{c,t} \cos \hat{\theta}_t}{(\hat{y}_{\ell,t} - \hat{y}_{r,t})^2} & \frac{\cos \hat{\theta}_t}{\hat{y}_{\ell,t} - \hat{y}_{r,t}} \\ \frac{\hat{y}_{r,t} \cos \hat{\theta}_t}{(\hat{y}_{\ell,t} - \hat{y}_{r,t})^2} - \frac{\hat{x}_{c,t} \sin \hat{\theta}_t}{(\hat{y}_{\ell,t} - \hat{y}_{r,t})^2} & \frac{-\hat{y}_{\ell,t} \cos \hat{\theta}_t}{(\hat{y}_{\ell,t} - \hat{y}_{r,t})^2} - \frac{\hat{x}_{c,t} \sin \hat{\theta}_t}{(\hat{y}_{\ell,t} - \hat{y}_{r,t})^2} & \frac{-\sin \hat{\theta}_t}{\hat{y}_{\ell,t} - \hat{y}_{r,t}} \\ \frac{-1}{(\hat{y}_{\ell,t} - \hat{y}_{r,t})^2} & \frac{1}{(\hat{y}_{\ell,t} - \hat{y}_{r,t})^2} & 0 \end{bmatrix}. \quad (9b)$$

3.1. ICR Estimation Using Terrain Adaptive IKF. First, the process model is established as

$$\dot{s} = g(s, v_{\ell}, v_r) + w, \quad (10)$$

where $s = [e, n, \theta, y_{\ell}, y_r, x_v]^T$. The specific form of (10) is

$$\begin{aligned} \dot{e} &= v_x \sin \theta + v_y \cos \theta + w_e, \\ \dot{n} &= v_x \cos \theta - v_y \sin \theta + w_n, \\ \dot{\theta} &= \omega + w_{\theta}, \\ \dot{y}_{\ell} &= w_{\ell}, \\ \dot{y}_r &= w_r, \\ \dot{x}_v &= w_v, \end{aligned} \quad (11)$$

where $w = [w_e, w_n, w_{\theta}, w_{\ell}, w_r, w_v]^T$ represents an additive Gaussian white noise vector. Each noise in w is unrelated to another. Previous research has demonstrated that the ICR locations are almost constant when moving on the same terrain, regardless of the robot manoeuvres [14]. Hence, the ICR locations can be modelled as a Markov-Gaussian stochastic process, that is, the ICR location at the current sampling point equals the sum of those at the last sampling point and an additive Gaussian white.

Furthermore, to facilitate the calculation in the digital devices, we derive the discrete-time form of (10) as

$$s_{t+1} = f(s_t, v_{\ell,t}, v_{r,t}) + T w_t, \quad (12)$$

where $s_t = [e_t, n_t, \theta_t, y_{\ell,t}, y_{r,t}, x_{v,t}]^T$ denotes the state vector and $w_t = [w_{e,t}, w_{n,t}, w_{\theta,t}, w_{\ell,t}, w_{r,t}, w_{v,t}]^T \sim \mathcal{N}(0, Q)$ denotes the process noise vector. The process noise variance $Q = \text{diag}(q_1, q_2, q_3, q_4, q_5, q_6) = \text{block diag}(Q_k, Q_s)$ where $Q_k = \text{diag}(q_1, q_2, q_3)$ and $Q_s = \text{diag}(q_4, q_5, q_6)$. The specific form of (12) is shown in (8), where T denotes the sampling period.

Second, to estimate s_t , the observation model is established as

$$z_t = H s_t + v_t, \quad (13)$$

where $z_t = [z_{e,t}, z_{n,t}, z_{\theta,t}]^T$ denotes the observation vector, $v_t = [v_{e,t}, v_{n,t}, v_{\theta,t}]^T \sim \mathcal{N}(0, R)$ denotes the observation noise vector which is assumed to be Gaussian white noises, and $H = [I_{3 \times 3}, O_{3 \times 3}]$ denotes the observation matrix where $O_{3 \times 3}$ and $I_{3 \times 3}$ denote a 3×3 null matrix and identity matrix, respectively. The observation can be realized by using a GPS-compass integrated sensor, which provides real-time measurements of robot's location and heading. Now s_t is estimated recursively by using the following estimation method.

3.1.1. Process Noise Variance Adjustment. Check λ consecutive recognized terrains provided by the Bayesian filter. If more than half of them are another terrain different from the current one, Q_s is set as $\text{diag}(q_{\max}, q_{\max}, q_{\max})$ where q_{\max} is a relatively large value, say 100. Otherwise, Q_s is set as $\text{diag}(q_{\min}, q_{\min}, q_{\min})$ where q_{\min} is a relatively small value, say 0.01.

The setup of λ is according to the classification accuracy. First, λ is an odd. Second, defining the minimum classification accuracy by p , the probability that more than $\lceil \lambda/2 \rceil$ of the λ recognized terrains are correct is

$$\rho = \sum_{i=\lceil \frac{\lambda}{2} \rceil}^{\lambda} \binom{\lambda}{i} \cdot p^i \cdot (1-p)^{\lambda-i}. \quad (14)$$

Let $p = 0.8$, then setting $\lambda = 3$ can guarantee $\rho = 0.9$. If $\rho = 0.98$ is desired, then λ must be larger than 9.

3.1.2. Priori Estimation. Calculate the a priori estimation \tilde{s}_t by

$$\tilde{s}_{t+1} = f(\tilde{s}_t, v_{\ell,t}, v_{r,t}), \quad (15)$$

and the corresponding error variance by

$$\tilde{P}_{t+1} = F_t \tilde{P}_t F_t' + L_t Q L_t'. \quad (16)$$

In (16), F_t and L_t denote the Jacobian matrices of $f(\cdot)$, that is,

$$\begin{aligned} F_t &= \left. \frac{\partial f}{\partial s_t} \right|_{s_t=\tilde{s}_t} = \begin{bmatrix} F_{\mathcal{K},t} & F_{\mathcal{S},t} \\ O_{3 \times 3} & I_{3 \times 3} \end{bmatrix}, \\ L_t &= \left. \frac{\partial f}{\partial w_t} \right|_{s_t=\tilde{s}_t} = T \cdot I_{6 \times 6}, \end{aligned} \quad (17)$$

where $O_{n \times n}$ and $I_{n \times n}$ denotes $n \times n$ null matrix and identity matrix, respectively. The specific form of $F_{\mathcal{K},t}$ and $F_{\mathcal{S},t}$ is shown in (9a) and (9b), respectively.

3.1.3. *Posterior Estimation.* Calculate the innovation δ_{t+1} by

$$\delta_{t+1} = z_{t+1} - H\tilde{s}_{t+1}, \quad (18)$$

and its covariance is estimated by

$$V_{t+1} = \frac{1}{N} \sum_{i=t-N+2}^{t+1} \delta_i \delta_i' = V_t + \frac{1}{N} \left(\delta_{t+1} \delta_{t+1}' - \delta_{t-N+1} \delta_{t-N+1}' \right), \quad (19)$$

where N denotes windowing size.

Furthermore, calculate the filtering gain K_t by

$$K_{t+1} = \tilde{P}_{t+1} H' V_{t+1}^{-1}. \quad (20)$$

Finally, the a posteriori estimation \hat{s}_{t+1} can be calculated by

$$\hat{s}_{t+1} = \tilde{s}_{t+1} + K_{t+1} \delta_{t+1}, \quad (21)$$

coupled with its error variance

$$\hat{P}_{t+1} = (I_{6 \times 6} - K_{t+1} H) \tilde{P}_{t+1}. \quad (22)$$

3.2. *Vibration-Based Terrain Classification.* The vibration is measured by an accelerometer mounted on the robot perpendicularly. The vertical acceleration time series describes the vibration caused by irregular terrain surfaces. First, experiments on different terrains should be conducted to constitute training samples. Second, we use a decision tree (DT) to learn the relationship between the vibration and terrain classes. Third, the classifier outputs are imported into an adaptive Bayesian filter, thus increasing the classification accuracy. The following sections will cover the feature extraction, DT classification, and Bayesian filter.

3.2.1. *Feature Extraction.* In [19–21], frequency domain-based features and deep features are employed in terrain classification. Using these features may improve the classification accuracy to some extent. However, their extraction algorithms are time-consuming and resource-intensive, which affects the real-time terrain classification implemented at low-performance digital computer. For example, in order to extract frequency domain-based features, the Fourier transform, which is relatively complex in computation, is an essential step. Hence, we extract features in time domain directly.

Between two successive sampling points, the robot vibration is measured at an accelerometer every T/n time, so n samples are obtained. Define $a_t = (a_t^{(1)}, a_t^{(2)}, \dots, a_t^{(n)})$ as the raw acceleration vector gathered from sampling point $t-1$ to t . Due to the presence of gravity, the accelerometer does not detect a pure vibration, but the vertical acceleration mixed with gravitational acceleration. Owing

to gravity G being relatively stable, which can be read as a DC component, a_t is preprocessed by

$$\hat{a}_t^{(i)} = a_t^{(i)} - G, \quad (23)$$

where $\hat{a}_t = (\hat{a}_t^{(1)}, \hat{a}_t^{(2)}, \dots, \hat{a}_t^{(n)})$ is the pure vibration vector. The feature vector $\phi_t = (\phi_t^{(1)}, \phi_t^{(2)}, \phi_t^{(3)})$ is extracted from \hat{a}_t , and its entries are shown as follows:

(i) $\phi_t^{(1)}$: the number of sign changes

$$\phi_t^{(1)} = \sum_{i=2}^n \zeta_i, \quad (24)$$

where

$$\zeta_i = \begin{cases} 1, & \text{if } \text{sign}(\hat{a}_t^{(i)} \cdot \hat{a}_t^{(i-1)}) < 0, \\ 0, & \text{if } \text{sign}(\hat{a}_t^{(i)} \cdot \hat{a}_t^{(i-1)}) > 0. \end{cases} \quad (25)$$

The case $\hat{a}_t^{(i)} = 0$ is ignored since it is singular. This feature is an approximation of the frequency of \hat{a}_t .

(ii) $\phi_t^{(2)}$: the mean of \hat{a}_t

$$\phi_t^{(2)} = \frac{1}{n} \sum_{i=1}^n \hat{a}_t^{(i)}. \quad (26)$$

Although the DC component of the raw data has been eliminated by (23), the mean of \hat{a}_t may considerably diverge from zero for some coarse terrain.

(iii) $\phi_t^{(3)}$: the variance of \hat{a}_t

$$\phi_t^{(3)} = \frac{1}{n} \sum_{i=1}^n \left(\hat{a}_t^{(i)} - \phi_t^{(2)} \right)^2. \quad (27)$$

Intuitively, the variance is higher when the terrain becomes coarser.

3.2.2. *DT Classification.* The decision tree (DT) is a frequently used machine learning method, the purpose of which is to establish a tree with high generalization ability. It is characterized by high classification efficiency and interpretability. A DT can be generated by using a ID3 algorithm. The core of a ID3 algorithm is to select features on each node of a DT by applying information gain, and then structuring the DT recursively. First, starting from the root node, calculate the information gain of all possible features on each node. Assign the feature with the highest information gain to the node, and then structure the child node based on different values of this feature. Second, apply the method in step

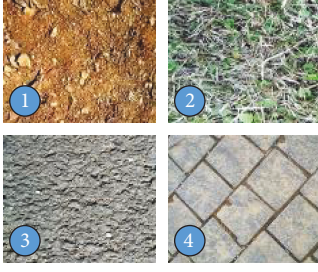


FIGURE 4: Photos of the traversed terrains.

1 to the child nodes recursively, until the information gains of all the features are small or there is no feature that could be selected, thus structuring a DT. The ID3 algorithm is similar to the probability model selection using the method of maximum likelihood.

The generated DT usually has high classification accuracy on the training data, but low accuracy on the test data, namely, overfitting occurs. This is due to an overcomplicated DT being trained during the learning process to fit the training data. Hence, it is necessary to simplify the DT to reduce its complexity, which is usually called pruning. Specifically, pruning means cutting some subtrees or leaf nodes from the generated DT, and treating their root or father nodes as new leaf nodes. The pruning is realized via minimizing the cost function. First, calculate the empirical entropy of each node. Second, retract upwards from each leaf node. Third, if the cost function value of the retracted DT is smaller than that of the unretracted DT, then conduct the pruning. Fourth, return to step 2 until it cannot be continued, thus obtaining a simplified DT.

3.2.3. Bayesian Filter. The recursive form of Bayesian filter can be seen in [22]. Define $\chi_t \in \{1, 2, \dots, k\}$ as the label of real terrain at sampling point t , $c_t \in \{1, 2, \dots, k\}$ as the classifier output, and $C_t = \{c_1, c_2, \dots, c_t\}$ as the measurement set. The purpose is to acquire $\mathcal{P}(\chi_t | C_t)$, the a posteriori possibility distribution function (pdf) of χ_t in the condition of C_t . Given $\mathcal{P}(\chi_{t-1} | C_{t-1})$, the a priori pdf of χ_t is

$$\mathcal{P}(\chi_t = i | C_{t-1}) = \sum_{j=1}^k \mathcal{P}(\chi_t = i | \chi_{t-1} = j) \mathcal{P}(\chi_{t-1} = j | C_{t-1}), \quad (28)$$

and the *a posteriori* pdf is calculated by

$$\mathcal{P}(\chi_t = i | C_t) = \frac{\mathcal{P}(c_t = j | \chi_t = i) \mathcal{P}(\chi_t = i | C_{t-1})}{\sum_{i=1}^k \mathcal{P}(c_t = j | \chi_t = i) \mathcal{P}(\chi_t = i | C_{t-1})}, \quad (29)$$

where $\mathcal{P}(\chi_t = i | \chi_{t-1} = j)$ denotes the probability that the robot moves from terrain j to i at time t and $\mathcal{P}(c_t = j | \chi_t = i)$ denotes the probability that the classifier outputs terrain j while the robot locates at terrain i . $\mathcal{P}(\chi_t | \chi_{t-1})$, which is needed during the time-update procedure, describes the intrinsic rules of terrain variation. Given k terrains, a $k \times k$ square matrix M with elements $m_{ij} = \mathcal{P}(\chi_t = i | \chi_{t-1} = j)$

TABLE 1: Confusion matrix of 10-fold cross-validation with all terrains.

	Soil	Grass	Asphalt	Paving
Soil	71	18	3	8
Grass	13	75	7	5
Asphalt	8	1	89	2
Paving	11	2	4	83

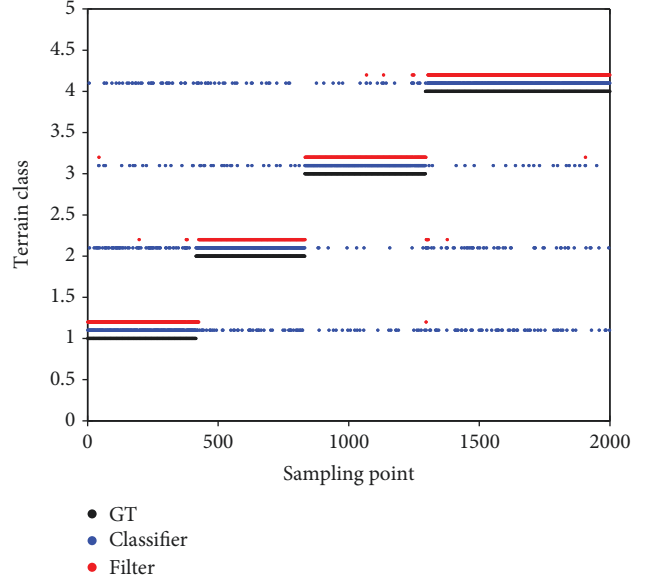
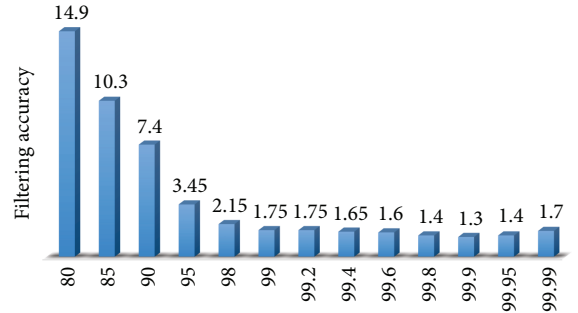


FIGURE 5: Filtering results. “GT” stands for the real terrain, “classifier” stands for the classifier output, and “filter” stands for the filter output.

FIGURE 6: Misclassification rate after using Bayesian filter under different μ .

should be defined. The diagonal elements m_{ii} where $i = 1, 2, \dots, k$ should be assigned a relatively large value not greater than 1. By assigning $m_{ii} = \mu$, a large value can enable most misclassified results to be corrected, but a succession of mistakes may appear after terrain variation. This is because m_{ii} should be reassigned a small value and m_{ij} should be relatively large at the moment that terrain changes from i to j . Hence, the terrain transition probability matrix should be dynamically adjusted when the terrain changes. In this

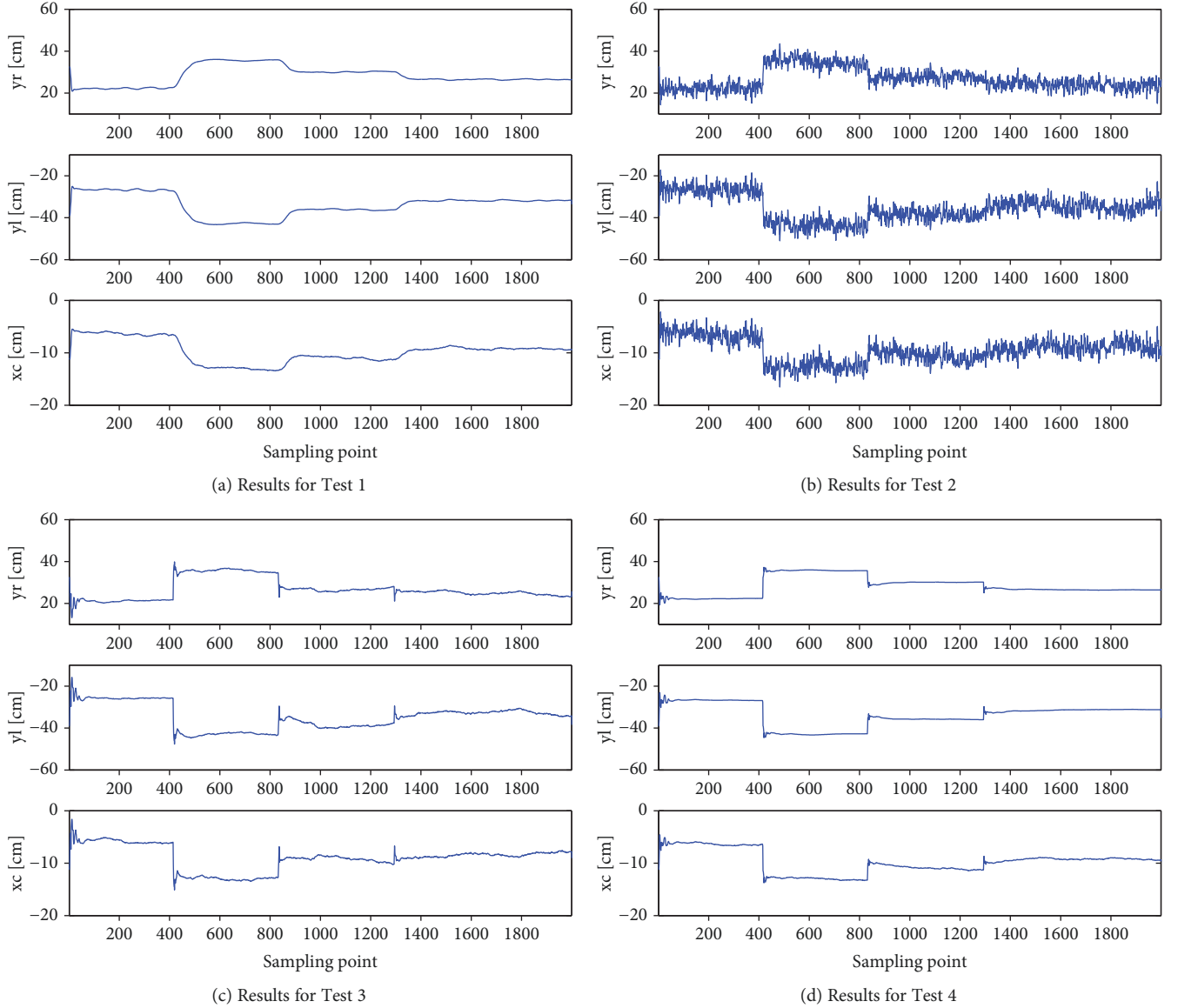


FIGURE 7: Results of ICR estimation test.

study, if terrain does not change, then V_t tends to be steady state; if the terrain changes, then a sudden change occurs in V_t . After detecting this event, we decrease m_{ii} and increase the related nondiagonal elements, and therefore, the accuracy of the Bayesian filter is increased.

4. Experiment and Analysis

In this section, a real-world experiment based on a tracked robot, equipped with two incremental encoders, a GPS compass integrated sensor and an accelerometer, is presented to verify the effectiveness of the proposed terrain adaptive IEKF. We mainly focus on the convergence rate and smoothness of the ICR estimation. Additionally, it is demonstrated that the accuracy of dead reckoning could be improved with knowledge of the ICRs.

The experimental robot was 260 millimeters in length, 210 millimeters in width, and 200 millimeters in height. Each

driving wheel was equipped with an incremental encoder of 540 resolutions. The GPS-compass integrated sensor was able to provide the robot's pose (including the position and heading) at 1 Hz with accuracies of 2 meters and 1 degree. As shown in Figure 4, during the experiment, the robot traversed soil, grass, asphalt, and paving successively. The accelerometer worked at the rate of 200 Hz, and the other sensors were read by the data collector at 1 Hz. After the data gathering, these data were sent to a computer (3.2 GHz with 8 GB RAM) and processed using MATLAB. It was noted that if the robot was equipped with a suspension system, or soft wheels, the vibration would be dampened or even absorbed. Hence, the accelerometer needed to be directly mounted on the robot chassis.

4.1. Terrain Classification. In order to collect the data for the classifier training, the tracked robot traversed the aforementioned four terrains at a speed of between 0.4 m/s

and 1.3 m/s. The test on each terrain lasted for about 1000 seconds, that is, 1000 samples were obtained for each terrain. We randomly divided these data into 10 parts on average and used each part to test the trained classifier once while used the remainder for training. The confusion matrix is shown in Table 1. Observing the diagonal elements, it can be seen that the classifier performs best at recognizing asphalt. This may be caused by the fact that most asphalt roads are flatter compared with soil, grass, and paving. When soil is exposed to the air, it turns dry and hard. After covering grass and then soil, the terrain becomes flatter and softer. However, the two terrains still share similar roughness, and therefore, the classifier performs worse at distinguishing soil and grass. It can also be observed that about one-tenth of the paving samples were misclassified as soil and vice versa. This is because soil and paving have the same hardness and roughness.

We use the trained DT classifier to recognize terrains when the robot is in actual operation. The classifier outputs are shown in Figure 5. Its misclassification rate is 20.39%, which cannot be acceptable. After using Bayesian filter, the misclassification rate can be decreased to a great extent. The filtering accuracy under assigning μ with different values is summarized in Figure 6. Observe that the misclassification rate can be reduced to a level under 2%. If V_t is introduced to adjust μ , the misclassification rate can be further reduced to a level under 0.6%.

4.2. ICR Estimation. The results of the ICR estimation test are shown in Figure 7. In Test 1 where $q_{\min} = 0.1^2$, as shown in Figure 7(a), the convergent tendency is obvious, but it took too much time to track the reality of the ICRs. If the terrain changes frequently, the ICR estimation may not track the truths. The results of Test 2 where $q_{\max} = 20^2$ are shown in Figure 7(b). The ICR estimation is able to converge rapidly, but not in a smooth manner. In Tests 3 and 4, by introducing terrain classification, the ICR estimation can track the truths rapidly when the terrain changes. After the estimation becomes stable, it does not oscillate around the truths significantly. In contrast to Test 3, IEKF is adopted in Test 4 to reduce the effect brought by uncertainties in the model and the measurements. Comparing Figures 7(c) and 7(d), TA-IEKF outperforms TA-EKF. A common method to verify the correctness of the ICR estimation is by terrain adaptive odometry (also known as terrain adaptive dead reckoning). As shown in Figure 8, we incorporate the ICRs obtained in Tests 1 to 4. Obviously, the TA-IEKF is able to provide the robot's ICRs with the highest accuracy, so the trajectory obtained from the terrain adaptive odometry is consistent with the outputs of the GPS, which has no cumulative errors.

5. Conclusion

In this paper, we propose an online estimation method to acquire robot's ICRs by means of data fusion technologies. First, an innovation-based extended Kalman filter (IEKF) is employed to fuse the readings from two incremental encoders and a GPS-compass integrated sensor, to provide a real-time ICR estimation. Second, a decision tree-based learning system is used to classify the terrains over which

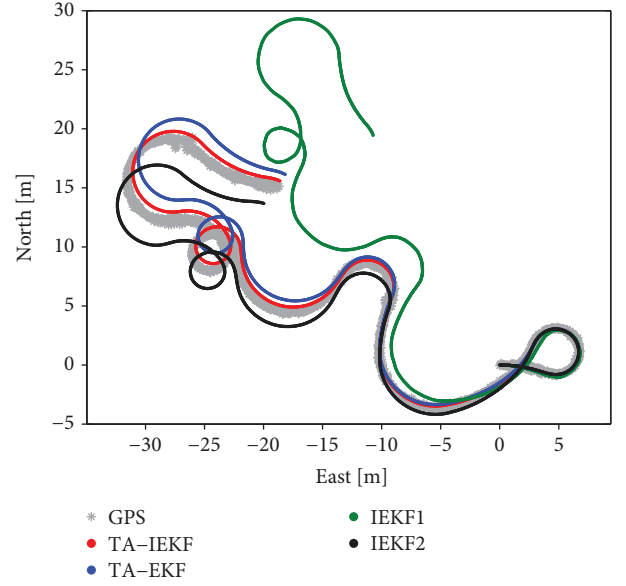


FIGURE 8: Results of dead reckoning with knowledge of ICRs. “GPS” stands for the outputs of GPS, “IEKF1” stands for the terrain adaptive odometry whose ICRs are obtained using IEKF with small Q , “IEKF2” stands for the terrain adaptive odometry whose ICRs are obtained using IEKF with large Q , “TA-EKF” stands for the terrain adaptive odometry whose ICRs are obtained using TA-EKF, and “TA-IEKF” stands for the terrain adaptive odometry whose ICRs are obtained using TA-IEKF.

the robot is traversing, according to the vibration signals gathered by an accelerometer. The results of the terrain classification are improved via a Bayesian filter, by utilizing temporal correlation in the terrain time series. Third, the performances of the ICR estimation and terrain classification are mutually promoted. On the one hand, terrain variation is detected with the aid of terrain classification, and therefore, the process of the noise variance of the IEKF can be automatically adjusted. Hence, the results of the ICR estimation are smooth on the same terrains and converge rapidly on terrain variation. On the other hand, sudden changes in the innovation are used to adjust the state transition probability during the recursive calculation of the Bayesian filter, thus increasing the accuracy of the terrain classification. The experiment indicates that the TA-IEKF-based ICR estimation has relatively higher accuracy, because it is able to converge rapidly at the moment the terrain changes and maintain the estimation smoothness when the robot is on the same terrain. Using the obtained ICRs to correct the odometry can retard its accumulation of localization errors, which is of great significance to the localization in GPS-denied areas.

Data Availability

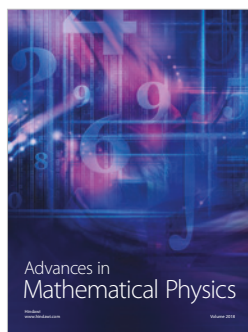
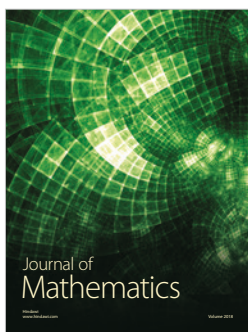
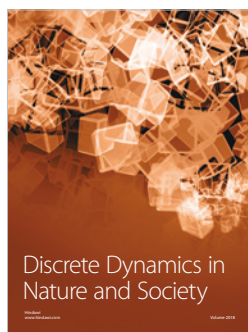
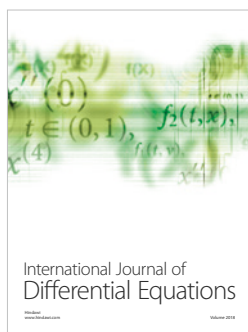
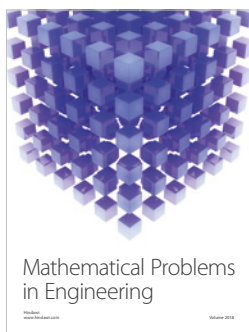
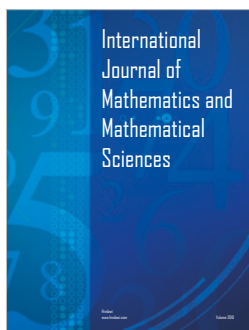
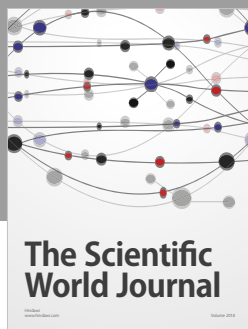
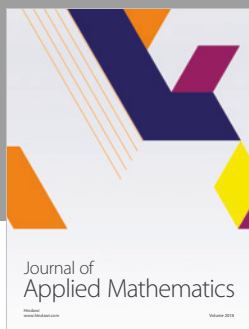
The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- [1] R. Siegwart, I. R. Nourbakhsh, and D. Scaramuzza, *Introduction to Autonomous Mobile Robots*, MIT press, 2011.
- [2] J. Yi, H. Wang, H. Zhang, D. Song, S. Jayasuriya, and J. Liu, "Kinematic modeling and analysis of skid-steered mobile robots with applications to low-cost inertial-measurement-unit-based motion estimation," *IEEE Transactions on Robotics*, vol. 25, no. 5, pp. 1087–1097, 2009.
- [3] M. Reinstein, V. Kubelka, and K. Zimmermann, "Terrain adaptive odometry for mobile skid-steer robots," in *2013 IEEE International Conference on Robotics and Automation*, pp. 4706–4711, Karlsruhe, Germany, May 2013.
- [4] J. Pentzer, S. Brennan, and K. Reichard, "The use of unicycle robot control strategies for skid-steer robots through the icr kinematic mapping," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3201–3206, Chicago, IL, USA, September 2014.
- [5] J. Pentzer, K. Reichard, and S. Brennan, "Energy-based path planning for skid-steer vehicles operating in areas with mixed surface types," in *2016 American Control Conference (ACC)*, Boston, MA, USA, July 2016.
- [6] J. Pentzer, S. Brennan, and K. Reichard, "On-line estimation of vehicle motion and power model parameters for skid-steer robot energy use prediction," in *2014 American Control Conference*, pp. 2786–2791, Portland, OR, USA, June 2014.
- [7] W. Yu, O. Y. Chuy Jr, E. G. Collins, and P. Hollis, "Analysis and experimental verification for dynamic modeling of a skid-steered wheeled vehicle," *IEEE Transactions on Robotics*, vol. 26, no. 2, pp. 340–353, 2010.
- [8] W. Lv, Y. Kang, and J. Qin, "Indoor localization for skid-steering mobile robot by fusing encoder, gyroscope, and magnetometer," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, no. 99, pp. 1–13, 2017.
- [9] J. L. Martínez, A. Mandow, J. Morales, S. Pedraza, and A. García-Cerezo, "Approximating kinematics for tracked mobile robots," *The International Journal of Robotics Research*, vol. 24, no. 10, pp. 867–878, 2005.
- [10] Y. Wu, T. Wang, J. Liang et al., "Experimental kinematics modeling estimation for wheeled skid-steering mobile robots," in *2013 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pp. 268–273, Shenzhen, China, December 2013.
- [11] T. Wang, Y. Wu, J. Liang, C. Han, J. Chen, and Q. Zhao, "Analysis and experimental kinematics of a skid-steering wheeled robot based on a laser scanner sensor," *Sensors*, vol. 15, no. 5, pp. 9681–9702, 2015.
- [12] A. Mandow, J. L. Martínez, J. Morales, J. L. Blanco, A. García-Cerezo, and J. González, "Experimental kinematics for wheeled skid-steer mobile robots," in *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1222–1227, San Diego, CA, USA, October–November 2007.
- [13] S. A. A. Moosavian and A. Kalantari, "Experimental slip estimation for exact kinematics modeling and control of a tracked mobile robot," in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 95–100, Nice, France, September 2008.
- [14] J. Pentzer, S. Brennan, and K. Reichard, "Model-based prediction of skid-steer robot kinematics using online estimation of track instantaneous centers of rotation," *Journal of Field Robotics*, vol. 31, no. 3, pp. 455–476, 2014.
- [15] T. M. Dar and R. G. Longoria, "Slip estimation for small-scale robotic tracked vehicles," in *Proceedings of the 2010 American Control Conference*, pp. 6816–6821, Baltimore, MD, USA, June–July 2010.
- [16] Y. Yao, K. Cheng, Z.-J. Zhou, B.-C. Zhang, C. Dong, and S. Zheng, "A novel method for estimating the track-soil parameters based on kalman and improved strong tracking filters," *ISA Transactions*, vol. 59, pp. 450–456, 2015.
- [17] X. Song, Z. Song, L. D. Seneviratne, and K. Althoefer, "Optical flow-based slip and velocity estimation technique for unmanned skid-steered vehicles," in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 101–106, Nice, France, September 2008.
- [18] Y. Li, L. Ding, and G. Liu, "Error-tolerant switched robust extended kalman filter with application to parameter estimation of wheel-soil interaction," *IEEE Transactions on Control Systems Technology*, vol. 22, no. 4, pp. 1448–1460, 2014.
- [19] J. Christie and N. Kottege, "Acoustics based terrain classification for legged robots," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3596–3603, Stockholm, Sweden, May 2016.
- [20] M. Cuneyitoglu Ozkul, A. Saranlı, and Y. Yazicioglu, "Acoustic surface perception from naturally occurring step sounds of a dexterous hexapod robot," *Mechanical Systems and Signal Processing*, vol. 40, no. 1, pp. 178–193, 2013.
- [21] A. Valada and W. Burgard, "Deep spatiotemporal models for robust proprioceptive terrain classification," *The International Journal of Robotics Research*, vol. 36, no. 13–14, pp. 1521–1539, 2017.
- [22] H. Fang, N. Tian, Y. Wang, M. Zhou, and M. A. Haile, "Non-linear bayesian estimation: from kalman filtering to a broader horizon," *IEEE/CAA Journal of Automatica Sinica*, vol. 5, no. 2, pp. 401–417, 2018.



Submit your manuscripts at
www.hindawi.com